A Macro-Level Order Metric for Self-Organizing Adaptive Systems

David W. King
Department of Electrical and Computer Engineering
Air Force Institute of Technology
Wright-Patterson AFB, Ohio 45433
Email: david.king@afit.edu

Gilbert L. Peterson
Department of Electrical and Computer Engineering
Air Force Institute of Technology
Wright-Patterson AFB, Ohio 45433
Email: gilbert.peterson@afit.edu

***Abstract*—Analyzing how agent interactions affect macro-level self-organized behaviors can yield a deeper understanding of how complex adaptive systems work. The dynamic nature of complex systems makes it difficult to determine if, or when, a system has reached a state of equilibrium or is about to undergo a major transition reflecting the appearance of self-organized states. Using the notion of local neighborhood entropy, this paper presents a metric for evaluating the macro-level order of a system. The metric is tested in two dissimilar complex adaptive systems with self-organizing properties: an autonomous swarm searching for multiple dynamic targets and Conway's Game of Life. In both domains, the proposed metric is able to graphically capture periods of increasing and decreasing self-organization (i.e. changes in macro-level order), equilibrium and points of criticality; displaying its general applicability in identifying these behaviors in complex adaptive systems.**

***Keywords*-self-organization; entropy; system order**

I. Introduction

Complex adaptive systems (CAS) are systems composed of a large population of agents that interact and adapt to their environment [1]. The aggregated interactions of these agents form complex behavior patterns [2] which appear to exceed the summation of those interactions [3]. In other words, the whole is greater than the sum of its parts. Historically, these emergent [4] or self-organizing [5] behaviors make these systems difficult to control [6]. Additionally, CASs pose significant challenges for researchers as they operate in dynamic environments with imperfect and incomplete information [7]. Engineered CASs are being fielded and interconnected every day [6], with this increased deployment comes an increased interest in developing methods for identifying possible self-organizing behaviors in complex systems. As agent interactions compose the driving force of a complex system's macro-level behavior [8], then tying those interactions to macro-level self-organizing patterns becomes the critical component for behavior analysis.

Whether it is a natural or engineered complex system, periods of stability and points of criticality occur during the system's lifetime. Points of criticality indicate a transition of the current system state to another, which may be a stable, unstable, or self-organized state [9]. For example, the collective rush of ants to gather a new food source will dominate the colony's short-term behavior. However, as the food dissipates, ants begin reverting back to other tasks such as brood sorting, nest defense and refuse cleaning; ensuring the colony's survival [10], [11]. Natural systems undergo periods of stability, e.g. a fairly even distribution of labor across all known tasks, and critical events, e.g. discovery of food creates an imbalance in ant task distribution. Langton [12] showed that periods of quiescence, as well as chaos, occur in cellular automata (CA), demonstrating that engineered CASs contain similar patterns to those found in natural systems. Cycles of stability in a CAS are indicative of a robust system as it is able to adapt to changes in the environment. Although prediction of such transitions and cycles is nearly impossible in real-time systems engaged in dynamic environments, it is possible to detect periods of stability and critical points in an *ad hoc* manner.

Inspired by entropy based approaches found in [13], [14] and newer research by [15], [16], this paper presents a macro-level metric for identifying periods of increasing and decreasing order. Order is the appearance and disappearance of self-organized behavior and indicates the stability and points of criticality in self-organizing systems. By coupling local entropy scores with agent distribution across neighborhoods, the metric captures macro-level agent dynamics, enabling detection of system stability and points of criticality. The metric is applied to both an autonomous Unmanned Aerial Vehicle (UAV) swarm engaged in a dynamic multi-target surveillance scenario and Conway's Game of Life. In both domains, the establishment of neighborhoods and population growth affect macro-level order. If neighborhoods remain static, i.e. the loss of population is minimal, the order of the system remains stable. When communities die off, or appear, noticeable departures in system order occur. Results show that the proposed metric is able to detect changes in system structure due to self-organized behaviors.

II. Related Work

Schulman and Seiden [13] first broached the idea of using system entropy, based on Shannon's entropy [17], as an indicator of order for Conway's Game of Life in

1977. More recently, Parunak and Bruecker [14] showed that directional entropy, a pheromone guidance approach for a walker searching for a target in unknown space, increases as walkers randomly move about, but reduces as agent knowledge (i.e. a walker encounters pheromones guiding it towards a target) increases, showing that information gain drives down the directional entropy of the system.

Wissner-Gross and Freer [15] and Mann and Garnett [16] take a slightly different approach. Instead of measuring micro-level entropy, they utilize macro-level entropy scores to guide micro-level agent decisions in complex systems. Wissner-Gross argue that causal entropic forces, based on maximizing system entropy, guide agents towards behaviors known as group intelligence (i.e. self-organization). In their view, agents independently choose actions that move a system towards macro-states with the highest causal entropy [15]. Their algorithm partitions the current system state into a set of possible future states and using Equation 1, calculates the entropic force (F) for that state.

$$F(X_0) = T\nabla_x S(X) \tag{1}$$

Here, T represents the reservoir temperature, $S(X)$ is the entropy associated with the macro-state X, and X_0, is the current macro-state. In this manner, one can evaluate multiple paths through possible future macro-states with agents choosing paths with the highest causal entropy. The main premise is that organized intelligence is due to individual agent decisions that result in future states that possess the most options for the agent (i.e. highest causal entropy). The paradox being that the organized behavior of these agents emerges although they collectively choose to increase the macro-level entropy of the system.

Mann and Garnett [16] extended Wisser-Gross and Freer's work by viewing future paths as independent Galton-Watson processes. This allows them to assign an exponential distribution to unknown future states, conditional on path extinction probabilities [16]. Again, consensus decision making (i.e. self-organized behavior), is the result of agents selecting maximum causal entropic paths. However, both approaches assume that: a)agents somehow know how to increase macro-level entropy and, b) possible future paths, and states associated with them, are countable, which is one of the difficulties in trying to use entropy measures in complex environments where state search spaces grow exponentially.

Shannon's definition of entropy [17] illuminates why entropy measures are interesting. Intuitively, if the amount of information discernible from a system grows, the system's macro-level entropy decreases. The correlation between a decrease in system entropy with the emergence of a pattern (i.e. order implies an information gain) makes Shannon's entropy equation a tempting approach for detecting self-organizing or emergent behaviors in a complex system. However, p_i in Shannon's equation (Equation 2) represents the probability of finding the system in state i [18]. Calculating every possible state for a system, especially in continuous environments, can quickly become computationally infeasible. It becomes even less practical in a real-time system where agents make decisions in timescales as small as microseconds.

$$H = -\sum_i p_i \log_2 p_i \tag{2}$$

One way to overcome this limitation is to discretize system spaces to reduce the number of possible system states. The benefit of such an approach is it allows one to expediently calculate Shannon's equation to measure system entropy. For example, although working in continuous space, Parunak and Brueckner [14] placed a grid over the system space, subsuming multiple states into separate partitions, allowing one to count the number of agents in those partitions which then define the system state. Mnif and Müller-Schloer [19] generalized the approach where one selects an attribute of the system that is discrete and enumerable. By doing so, one focuses on attributes of interest in which entropy measures carry more precise meaning while reducing the number of system states, making the use of entropy measures tractable. However, as [14] points out, one must couple micro and macro-level entropy in order to capture the imposition of order upon the system.

Parunak and Brueckener's idea is modeled after emergence researchers such as Holland [4] who argue that macro-level patterns rely on changing micro-patterns and it is through the aggregation of simple agent interactions that higher-level emergent behaviors arise [8]. What distinguishes emergent behavior at the macro-level is the absence of those behaviors at the micro-level [20]. The agents at the micro-level contain no knowledge beyond their immediate environment, or 'niche' [2]; however, their aggregated actions create complex patterns in their *niche* which then aggregate to create complexity in the ecosystems that subsume them. Agent knowledge is key here, unlike [15] and [16], global knowledge for agents is not realistic. Self-organized, or emergent patterns, develop from limited local agent knowledge. The autonomous swarm introduced in Section III will show how agents with limited knowledge can create recognizable, global patterns by choosing actions leading to the highest local entropy state. As macro-level entropy changes may indicate some type of order change or self-organized behavior [5], one must consider the accumulation of local entropy at the system level. However, mere averaging or summation of local entropy fails to convey much meaning about the type of order or behavior that is occurring and, in some cases, could fail to detect self-organized behavior [9]. In order to capture more information about agent interactions, this work proposes to tie both micro-level entropy and macro-level population density together to achieve a metric of system order that can also detect periods of stability and points of criticality.

The metric is based on Bonabeau, et al.'s threshold function (Equation 3), introduced as a model of task division in insect societies [21]:

$$T_\theta(S) = \frac{s^n}{s^n + \theta^n} \tag{3}$$

In their equation, the probability of performing a task is based on a response threshold, θ, related to a stimulus, s, associated with a task, and, n, determines the steepness of the threshold. Accordingly, if $s << \theta$, the probability of performing the task is almost 0, while if $s >> \theta$, then it is close to one [21]. This type of interplay between the stimulus, s, and threshold, θ, is exactly the type of behavior one needs to capture between micro and macro-level behaviors. Section III shows an extension that provides a metric of macro-level order with enough sensitivity to produce a reasonable measure of system order that indicates both times of periodic stability and points of criticality. As the metric is tied to population density, system order also provides general information about the number of neighborhoods and the distribution of the agent population across them.

Using Mnif and Müller-Schloer's [19] generalization approach, system states are measured as the number of agents assigned to specific neighborhoods, making state probability calculations tractable, enabling the use of entropy measurement. For this research, agents in the UAV domain continuously seek to maximize the entropy value of their local neighborhood. The constant pursuit of entropy maximization results in self-organized teams with shared common tasks. In the Game of Life, local entropy values are evaluated but do not influence agent decisions. However, coupling micro-level entropy and population distributions to macro-level order enables the identification of phase transitions (e.g. group emergence, self-organization) in the system. In both domains, local entropy scores and population distributions across neighborhoods produce an order measurement that is sensitive to micro-level dynamics.

III. Approach / Methodology

This work presents a metric that uses local entropy measures as a part of calculating macro-level order. As domains change, local entropy measures necessarily require different calculations. However, the macro-level order metric only needs the local entropy values. In other words, at the macro-level, the metric relies on one's definition and implementation of local entropy. The macro-level order metric is defined by:

$$\tau_{macro} = \frac{1}{|N|} \sum_{n=1}^{|N|} \frac{S(n)^2}{S(n)^2 + {\Delta_n}^2} \tag{4}$$

N is the set of all neighborhoods, $\{n_1, n_2, ..., n_{|N|}\}$, present in the system. S(n) is the local entropy associated with neighborhood, n, and Δ_n is the number of agents in that neighborhood divided by the total number of agents. Macro-level order is measured by a summed average of local neighborhood entropies, $S(n)$, and the macro-level distribution of agents, Δ_n, normalized by $|N|$. By using this equation, macro-level measurements carry indications of what is happening at the micro-level. For example, if the number of agents associated with a neighborhood, Δ_n, increases and begins surpassing $S(n)$, macro-level order decreases. This indicates an imbalance of agent distribution. For example, a larger percentage of the agent population lives in one area. If the sum of the local entropies of all agents begins to dominate, Δ_n, then macro-level order increases towards 1. Higher values indicate the emergence of multiple self-organized neighborhoods. Periods of equilibrium occur when the agent population is fairly evenly distributed (i.e. $S(n) \approx \Delta_n$) and remains relatively constant.

As recommended by a critique on the application of metrics across domains [9], the proposed macro-level order metric is tested in two dynamic domains: a UAV swarm engaged in a multi-task environment and the Game of Life.

The first domain of study consists of a swarm of UAVs engaged in intelligence, surveillance and reconnaissance (ISR) missions. When the simulation starts, all UAVs fly randomly through a bounded grid (1,000 x 1,000 pixels). Random targets, modeled as tasks, are added to the simulation at random locations and times. Any target (i.e. task) within range of a UAV sensor is added as a task to its task list. Agents calculate their local entropy by polling neighboring agents to find the distribution of agents across known targets. The agent proceeds to add itself to the number of agents assigned to task 1, and calculates its entropy using Equation 5 but instead of A_T equalling the number of agents in the entire simulation, it equals the number of agents in the agent's neighborhood. It continues the calculation for each known task, selecting the target that yields the highest local entropy.

Targets possess a limited, random, lifespan and disappear once the time limit is exceeded. Once a target disappears, agents must decide on a new target to fly towards, or revert back to random flight if no other targets exists within their sensor range. Figures 1-2 are screenshots from one simulation run. Agents (triangles) detect three distinct targets (circles). Guided by maximizing local entropy, they self-organize into three teams to cover each target.

The problem facing the agents is multi-fold. One, agents must decide what target, possibly among many, they must

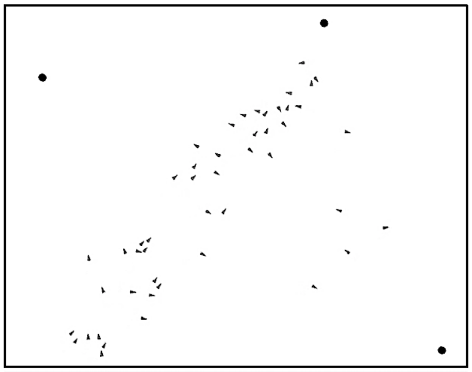

Figure 1. Three targets (circles) detected by the swarm (triangles).

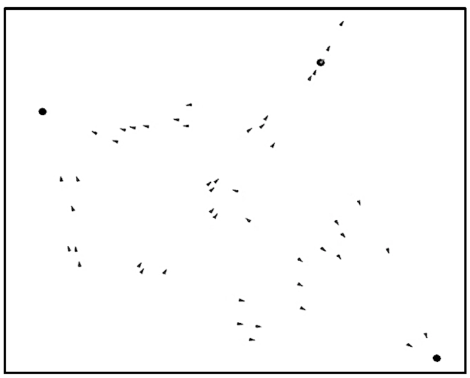

Figure 2. Swarm self-organizes into three teams which begin flying towards selected targets.

engage. Additionally, all decisions are based solely on local information. They can query agents within range about their current actions, but do not have global information on the location of agents outside of communication range or targets outside of sensor range. Finally, target appearances occur in a dynamic fashion. Agents must pursue new targets as they are detected, leading to dynamic task switching and coordination with other agents. However, coordination is limited to known information. Agents do not assume leadership roles nor do they task other agents. Each agent decides their task independently. Also lacking is a central controller. Once the UAV swarm is launched, all agents act autonomously.

For an autonomous UAV swarm in a dynamic, multi-task environment, task entropy is defined as:

$$S(n)_{uav} = -\sum_{i} \frac{A_i}{A_T} \log \frac{A_i}{A_T} \tag{5}$$

where A_i equals the number of agents assigned to task i and A_T equals the number of agents in the simulation. Using Equation 4, the macro-level order metric, τ_{macro}, is:

$$\tau_{macro} = \frac{1}{|N|} \sum_{i} \frac{S(n)^2_{uav}}{S(n)^2_{uav} + (\frac{A_i}{A_T})^2} \tag{6}$$

For the UAV domain, macro-level order is measured by the normalized sum of task entropies, $S(n)$, the percentage of agents assigned to task i, $\frac{A_i}{A_T}$, divided by the total number of tasks, $|N|$. With these measures, agents switching between tasks create noticeable changes in the macro-level order in both directions. When agents discover new tasks, macro-level order increases, while, as tasks dissipate or agents remove themselves from tasks, macro-level order decreases, yielding indicators of micro-level behaviors at the macro-level. When no tasks are available, macro-level order is simply 0, as agents revert to random search flight patterns.

The second domain is Conway's Game of Life. The basic concept for the Game of Life is a 2D grid where cells are either alive or dead. Depending on the number of neighbors (8 adjacent cells) a cell possesses, then it will either die from overcrowding or isolation. Dead cells may become alive if they possess three neighbors. From these simple rules, complex patterns of oscillation and change occur. Some patterns, such as *gliders*, appear to move across the screen while others, like the Gosper gun, continuously create other structures. For this paper, the environment is bounded by a 100 x 100 grid. This bounding stops gliders from perpetuating forever, resulting in a solid block structure on the edge. What is interesting is trying to identify periods of oscillation, which can indicate periods of stability, or transformation, as the case with a glider colliding with a wall.

In the Game of Life, local entropy must be calculated differently as agents here do not move anywhere and their decisions are based on a set of rules. Any type of motion associated with the Game of Life is placed upon the system by the observer, making the appearance of 'movement' a self-organized behavior that is the result of cells birthing and dying. Schulman and Seiden [13] analyzed various statistical properties of the Game of Life and proposed an entropy measure based on the size of a grain of cells and a proportion of the living cells compared to the expected average (Equation: 7). In this manner, they created a probability distribution for the living cells in the game. Instead of setting a set grain size (e.g. 10 x 10 square), the current state of the game is viewed as a graph, G, where every cell is a vertex, V. Each connected subcomponent, δ_s, in G is treated as a grain.

$$S(n)_{life} = -\frac{j^2}{J^2} \sum_{1}^{|\delta_s|} [\frac{A_{ni}}{j^2} log \frac{A_{ni}}{j^2} + (\frac{j^2 - A_{ni}}{j^2}) log (\frac{j^2 - A_{ni}}{j^2})] \tag{7}$$

Here, A_{ni} is the number of nodes in subcomponent, δ_{si}, j is the total number of alive cells, and J is the size of the grid (e.g. 100 x 100, $J = 100$). The state of the game at time step, t, is treated like a graph, G, where each community is a connected sub-component, δ_s, of G. These calculations feed into the macro-level order metric:

$$\tau_{macro} = \frac{1}{|\delta_s|} \sum_j \frac{S(n)^2_{life}}{S(n)^2_{life} + (\frac{|\delta_{si}|}{j})^2} \tag{8}$$

Instead of dividing neighborhoods up by task, neighborhoods in the Game of Life are divided into connected sub-components, δ_s. As with the UAV domain, changes in the distribution of *live* cells changes the macro-level order of the system. When very few sub-components exists, the system order is low, while increases in sub-components creates higher macro-level order. The one drawback for the metric with respect to the Game of Life is the appearance of one subcomponent that never changes in size (i.e. population remains constant) but its position changes. For example, a glider moving in continuous space would never change the macro-level order of the system. However, in bounded space, the glider will eventually collide with a wall and transform into a block of four living cells, changing the macro-level order of the system.

IV. Experimental Analysis

A. Autonomous UAV Simulation

The first set of experiments focused on the autonomous UAV swarm utilizing the local maximum entropy decision process. These experiments tested two questions: one, could maximizing local entropy yield complex, macro-level patterns aimed at task accomplishment, and two, does the macro-level order metric indicate periods of system stability and points of criticality. For each scenario, 50 UAV agents were assigned to the available labor pool. The population was homogeneous with each agent possessing the same capabilities, sensor ranges and max fuel capacity. Only initial positions and fuel levels were randomized. In order to refuel, an agent had to disengage from its current task and return to base, creating imbalances in target coverage. Beginning at time step 1,000, up to two targets appear, each at a 45 percent probability.

1) Scenario 1: In this scenario, agent sensor and communication ranges covered the entire 1,000 x 1,000 pixel grid, enabling complete knowledge of available tasks as well as allowing agents to coordinate with all other agents in the simulation (see Table I). Starting at time step 1,000, targets appear and disappear with a 45 percent chance, and at every 500 time steps thereafter. This scenario shows how agents maximizing local entropy creates self-organized teams and how changes in team membership (i.e. number of agents covering a task) create noticeable macro-level disturbances.

As a plot of labor distribution over time becomes unwieldy beyond one or two tasks, Table II contains labor distribution values at possible critical points (i.e. tasks appear or disappear). Figure 3 plots the macro-level order of the system over time. The initial jump in order occurs when the first two tasks appear, followed by a stable period of task coverage. At 1,500, a drop in order occurs as one task is removed and replaced by another. At 2,000, all tasks disappear from the simulation. At 3,000, a spike occurs due to a balanced distribution of task-assigned and refueling agents, leading to a period of stability, even when a task disappears and is replaced by another at 3,500. This occurs as agents cycle between refueling and encountering tasks afterwards. A noticeable jump occurs at 4,500 when a third task appears and labor distribution remains fairly even.

These result show that maximizing local entropy does lead to the creation of self-organized behaviors. Agent decisions create fairly balanced teams across known system targets. These self-organized teams create periods of stability easily seen in the order graph. In addition, points of critical system transition, both to higher and lower levels of system order, are collected. Although possible transition points were known *a priori*, they were dynamic where the appearance

Table I
Control and Independent Variables for UAV Scenario 1. All speeds are in pixels per time step, with ranges in pixels. Fuel and life metrics decremented by 1 at each time step.

Control			**Independent**	
Refueling Center	(500, 950)		Fuel Level	[3,000, 8,000]
Visual Radius	360 degrees		Sensor Range	1,000
Velocity	0.50		Comm Range	1,000
Swarm Size	50		Target Life	[500, INF]

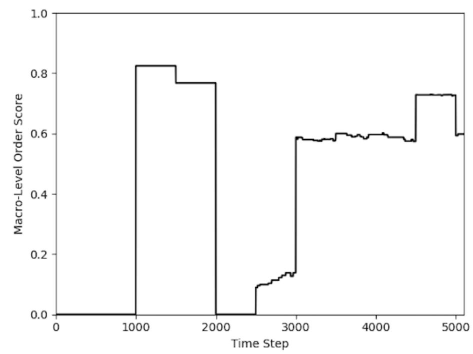

Figure 3. Macro-level order score over time.

Table II
Number of Agents per Task (Scenario 1)

		Time Step						
		1000	**1500**	**2000**	**2500**	**3000**	**3500**	**4500**
Task	**1**	25						
	2	25	25					
	3		25					
	4				44	11		
	5					22	16	10
	6						20	
	7							16
	8							15
	NONE			50	6	17	14	9

Table III
CONTROL AND INDEPENDENT VARIABLES FOR UAV SCENARIO 2. ALL SPEEDS ARE IN PIXELS PER TIME STEP, WITH RANGES IN PIXELS. FUEL AND LIFE METRICS DECREMENTED BY 1 AT EACH TIME STEP.

Control			**Independent**	
Refueling Center	(500, 950)		Fuel Level	[3,000, 8,000]
Visual Radius	360 degrees		Sensor Range	500
Velocity	0.50		Comm Range	250
Swarm Size	50		Target Life	[500, INF]

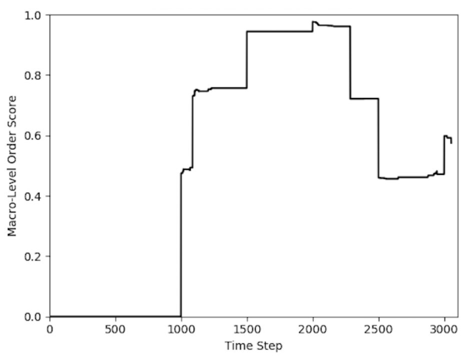

Figure 4. Macro-level order score over time.

Table IV
NUMBER OF AGENTS PER TASK (SCENARIO 2)

		Time Step						
		1000	**1500**	**2000**	**2280**	**2500**	**2750**	**3000**
Task	**1**	24	17					
	2	24	16	14	12			
	3		17	9	18	37	36	36
	4			16	19			
	5					8	11	2
	6							3
	NONE	2		11	1	2	3	9

or disappearance of targets was based upon random chance. The drop at time step 1,500, captures a change in local agent entropy calculations as one target disappeared as a new one appeared. Although the resulting distribution was the same, i.e. even distribution across two tasks, the metric captured the dynamic change which would have been missed by pure distribution metrics. Finally, this graph shows how the order metric picks up on other agent behaviors, such as refueling. Gradual decline or increase in system order due to agent refueling is captured during regions of relative stability.

2) Scenario 2: In this scenario, agent sensor and communication ranges were limited to 500 and 250 pixels respectively, increasing the likelihood of uneven task distribution in the system (see Table III). All other settings such as possible transition points and fuel randomization remained the same.

Table IV and Figure 4 show task distribution and macro-order scores over time for the second scenario. As before, when there is a fairly even distribution of labor across multiple tasks, the macro-level order increases towards 1. Additionally, the macro-level order drops when tasks disappear and agents reassign themselves. The system reaches

Table V
CONTROL AND INDEPENDENT VARIABLES FOR UAV SCENARIO 3. ALL SPEEDS ARE IN PIXELS PER TIME STEP, WITH RANGES IN PIXELS. FUEL AND LIFE METRICS DECREMENTED BY 1 AT EACH TIME STEP.

Control			**Independent**	
Refueling Center	(500, 950)		Fuel Level	[3,000, 8,000]
Visual Radius	360 degrees		Sensor Range	500
Velocity	0.5		Comm Range	500
Swarm Size	50		Target Life	[500, INF]

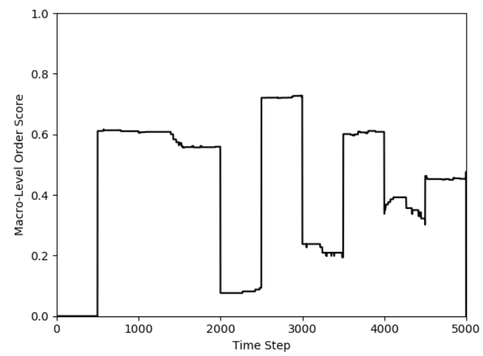

Figure 5. Macro-level order score over time.

a peak macro-level order score at 2,200, when the agents disperse evenly across three known tasks plus an equal number engaged in refueling. At 2,280, a sudden drop occurs as the refueled agents discover and engage tasks in the environment. This time step shows how sensitive the metric is to changes in system structure. At 2,500, there is another drop due to the loss of a task, resulting in large imbalance. As stated earlier, this occurrence will be fairly common when agents have restricted communication ranges. Finally, macro-level order appears relatively stable until 3,000, when Task 6 appears, resulting in a change in agent distribution.

Scenario 2 again highlights that agents seeking maximum local entropy scores creates self-organized behaviors that place order upon the system. Furthermore, the uneven distribution of agents across tasks creates a more chaotic, or less orderly system, reflected by lower order scores associated with time steps where the population of agents was skewed across targets. However, the order metric is still able to identify periods of stability and points of criticality.

3) Scenario 3: In the final UAV scenario, possible target appearance and dissolution occurred every 500 time steps at a 65 percent probability, creating a highly dynamic environment. All UAV sensor ranges were set to 500 pixels (see Table V).

Table VI and Figure 5 show task distribution and macro-order scores for the third scenario. Clear periods of stability and points of criticality appear in Figure 5. Starting at time step 500, the system enters into a period of relative stability as agents equally disburse across two tasks and random searching. A gradual decline at 1,500, occurs when agents engaged in random search begin finding tasks in

Table VI
NUMBER OF AGENTS PER TASK (SCENARIO 3)

Task \ Time Step	500	1000	1500	2000	2500	3000	3500	4000
1	17							
2	17							
3		17						
4		17						
5			26					
6			21					
7				46	13	21		
8					17			
9					15			
10							18	24
11							21	
12								
13								
NONE	16	16	3	4	5	29	11	26

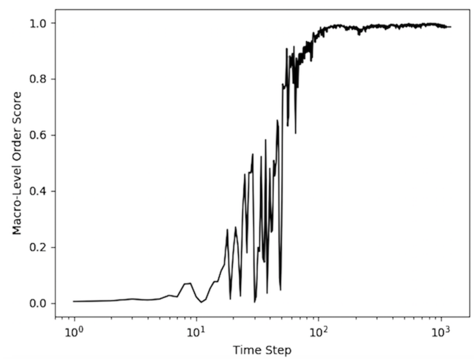

Figure 6. Macro-level Order for R Pentomino.

the environment. A sharp drop in system order occurs at 2,000, as the system reduces from two tasks to one that attracts a large proportion of the agent population. After time step 2,000, increases and decreases in system order (i.e. points of criticality) align with known checkpoints for possible task creation or dispersement (every 500 time steps) with periods of stability appearing between these points. Small perturbations in system order that occur during these stable regions are due to small numbers of agents switching from random search or refueling to target engagement, or vice versa, with rises associated with agents disengaging from targets and dips associated with agents finding, and engaging, them. In this manner, both random search and refueling create another group structure in the simulation that impacts system order.

B. Game of Life

The Game of Life is more stochastic than the UAV swarm with new cells being created and destroyed at almost every time step. Like the UAV simulation, periods of stability occur, usually as a result of the emergence of an oscillating pattern. Points of criticality, however, are more difficult to detect unless the system experiences major changes in neighborhood structures. For example, a glider moving across will not change the system order. From the order metric's perspective, the system is in a constant or quiescent state.

1) Scenario 4 – R Pentomino: R-Pentomino was selected for its chaotic start providing a challenge for the system order metric as points of criticality may be obfuscated by the noise of constant neighborhood creation and destruction in the early time steps.

The macro-order graph (Figure 6) shows that although the R Pentomino pattern immediately enters a highly stochastic state, the order of the system, and number of stable neighborhoods, increases during the first 200 time steps. Then the system undergoes a long period of stability with minor perturbations occurring as the pattern plays out. Many of the early neighborhoods enter quiescent states with the noted changes occurring when new neighborhoods go through their lifecycle. At time step 1,130, the entire system enters a quiescent state.

Although able to track these rapid changes in structure and population size, the order-metric cannot pinpoint critical transition points with any amount of certainty. If one averaged every 10 time steps into an order value, only one period of transition, time steps 10 - 30, creates a negative growth state (i.e. the order decreases). The rest of the time, the system is increasing in order until about time step 200. Arguably, every time step in the early life cycle of the R Pentomino pattern is a critical point, only stabilizing once larger, stable, neighborhood structures appear.

2) Scenario 5 – Glider: This work selected the glider pattern as it thwarted one of the entropy based metrics in [9]. In this scenario, the glider pattern moves across the screen until impacting the boundary and creating a permanent block structure. The macro-level order metric (Figure 7) clearly shows the boundary impact point at time step 230 and the resulting, permanent, block pattern from 231 onwards. As stated earlier, the order metric does not denote any difference in the glider's state until it comes into contact with the boundary and undergoes a transformation in size. The metric would have to be modified to be able to discern differences in system state due to glider movement where the structure composition fails to change (i.e. same number of connected alive cells) but the positions of the alive cells change. However, unlike the R Pentomino pattern, clear, critical transition points exist in the glider pattern, as indicated by the large drop in system order. The resultant shape, a four block square, is smaller than the original structure, and thus imposes less order on the entire system.

3) Scenario 6 – Gosper Gun: The Gosper gun is interesting as it has multiple subcomponents whose transformations continuously create gliders moving in a southwest direction from the "gun" group. In other words, a stable pattern continuously creates new patterns *ad infinitum*. Graphing macro-level order values (Figure 8) reveals some interesting patterns. First, the initial, stable patterns show clear oscillation features as they transform states. These transforming

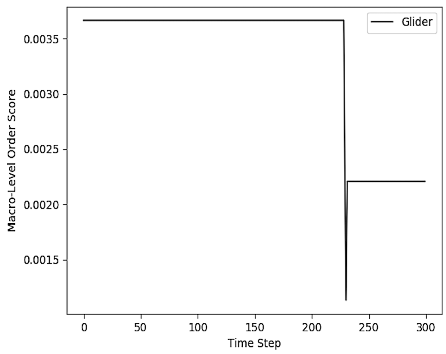

Figure 7. Macro-level Order for a Glider.

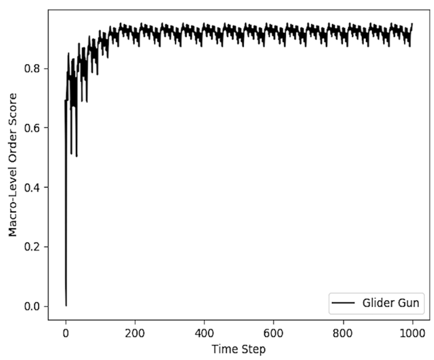

Figure 8. Macro-level Order for a Gosper Gun.

states then create a greater oscillation pattern as they produce gliders. The ramp up in glider production levels reaches its highest peak near time step 200. Afterwards, the entire system is an oscillating pattern of glider production and destruction as they impact the right most wall. The dips in the oscillation pattern represent the transformation and destruction of gliders against the right wall. A glider at time step, n, will collide with the wall and transform into a four block neighborhood. The glider at time step, $n + 2$, will then collide with the four block neighborhood, resulting in the 'death' of the cells and a drop in macro-level order. The next glider in formation will start the pattern all over again. Again, the order-level metric can detect changes in system order due to structural change and track periods of stability. However, critical points are still difficult to pinpoint in the early stages of system macro-order growth.

V. Discussion

These results highlight some interesting points. First, agents maximizing local neighborhood entropy can impose order on the system through self-organizing behaviors, implying agents do not require global knowledge to create an adaptable system. This observation is in-line with what is known about decentralized natural systems. Ants and bees have limited memories and sensors [22], [23], basing decisions upon current environmental stimuli which includes input from other agents. Some ants species in particular have shown a threshold based response where an ant's action decision is impact not by pheromones but by the number of times it encounters another type of ant [23]. In other words, evolution tuned ant behaviors to reward a division of labor but task selection is more intricate than strong pheromone trails and may rely on some clock that tiggers upon some set proportion. Bees engage in similar behaviors; however, bees use more complicated signaling methods, such as the waggle dance [22], as airborne pheromones would not serve their purpose very well. Gifting agents in a system full global knowledge or the ability to guesstimate millions of future states is both contrary to observation and, as this work shows, unnecessary to produce intelligent, collective behavior.

Additionally, local neighborhood entropy and population distribution produce a macro-level order metric sensitive to micro-level dynamics incorporating agent interactions, e.g. communication in the UAV swarm, and structure, such as connected cells in the Game of Life. It also takes into account temporal dynamics without having to calculate them explicitly. For example, the jagged periods of stability in the UAV swarm correlate to agent refueling behaviors tracked by task distribution versus individual fuel levels or a forced refueling function. The critical points seen in the UAV graphs also highlight the appearance and disappearance of tasks. Although coded to occur at certain time steps, the macro-level order metric is sensitive to changes in the environment resulting in observable deviations from current agent behaviors and neighborhood structure. One could leverage this in a detection scheme where a sliding scale average is used. Any value that is two to three standard deviations from the current window mean would alert an observer to a possible change in system behavior. These changes could be desired self-organized behaviors, such as teams forming for target coverage, or possible malicious acts. Using the UAV example, if the expected behavior is the even distribution of UAVs across known tasks and yet multiple UAVs continue to fly in random patterns or converge to one specific target, this could indicate some type of control failure or hijacking.

The central thread for tying micro and macro-level behaviors together is assigning meaning to the entropy metrics used. Here, macro-level order indicates the formation, or dissolution, of agent structures and the population assigned to them. If the macro-level order rises, teams are forming, if the macro-level order decreases, teams are dissolving, making the metric useful for interpretation. Additionally, reducing system states to a generalized attribute [19] allows the use of entropic methods without having to estimate possible probability distributions or densities. Attribute-based entropy methods allow for specific probability measurements.

However, limitations for this approach do exist. One, not all systems can be easily reduced to a set of even priority tasks with homogeneous agents or neighborhood structures. For heterogeneous agents, one would need either separate entropy measures or some type of hybridized approach to estimate the global macro-order related to each type of agent. Otherwise, some underlying micro-pattern changes may be lost in the "noise" of a combined estimation. As Haghnevis [24] argues, simply combining three components with entropy values, $x + y + z = 1$, tells one very little as any of the components could hold those particular values. Furthermore, task prioritization is ignored in this method. It is possible to give tasks a higher priority but then a threshold parameter needs to be introduced to enforce division of labor at some point. Without this threshold, all available agents would flock to the high priority task, ignoring all others, which is probably an undesirable feature. Finally, this approach does not guarantee team-to-task optimality. The agents select tasks based solely upon local entropy levels without regard to current fuel status or distances, leaving some tasks untended as the agent that selected that task returns to refuel and in some cases, the target dissipates before the agent arrives when another agent could have reached it sooner.

VI. Conclusion and Future Work

This paper presented a macro-level order metric that incorporates both local entropy scores and macro-level agent population distributions. Experimental results showed that the macro-level order metric is able to detect both periods of stability and points of criticality in two dissimilar domains. The estimated system order reflected the state of self-organized behaviors in the system as well as enabling detection of stability and critical points during the system's lifecycle. Although an exploratory method, the metric showed sensitivity to micro-level agent interactions on a scale detectable to an outside observer. In this manner, this metric can be used for the detection of emergent patterns of self-organization and decomposition inside a complex system. This work showed that micro-level entropy scores correlate to increasing and decreasing macro-level order imposed upon the system that align with Holland's hierarchical view that macro-level patterns rely on changing micro-level patterns [4] in complex adaptive systems.

Future work will need to incorporate different entropy measures obtained from a heterogeneous population of agents. Additionally, ideas such as task prioritization and response thresholds should be added to analyze the affect those changes have upon self-organized behaviors. These additions would necessitate the development of a modified macro-level order metric that could account for the various types of micro-level entropies and neighborhood distributions. The current metric would account for the overall order; however, causation of gain and loss may be lost as different agent interactions are rolled into one. Finally, an obvious extension is to apply this metric to other domains. One particular domain could be an ant-colony simulator to see if entropy based results reflect real-world ant behaviors.

References

[1] J. H. Holland, "Studying complex adaptive systems," *Systems Science and Complexity*, vol. 19, no. 1, pp. 1–8, 2006.

[2] ——, *Signals and Boundaries: Building Blocks for Complex Adaptive Systems*. MIT Press, 2012.

[3] D. Pais, "Emergent collective behavior in multi-agent systems: An evolutionary perspective," Ph.D. dissertation, Princeton University, 2012.

[4] J. H. Holland, *Emergence: From Chaos to Order*. Addison-Wesley, 1998.

[5] I. Breddin, "Self-organisation and emergence," Technical University of Berlin, Tech. Rep., 2006.

[6] M. Couture, "Complexity and chaos – state-of-the-art; overview of theoretical concepts," DRDC Valcartier TM, Tech. Rep. 2006-453, 2007.

[7] S. Ontanon, G. Synnaeve, A. Uriarte, F. Richoux, D. Chruchill, and M. Preuss, "A survey of real-time strategy game ai research and competition in starcraft," *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 5, no. 4, 2013.

[8] S. Rasmussen, N. A. Baas, B. Mayer, and M. Nilsson, "Defense of the *ansatz* for dynamical hierarchies," *Artificial Life*, vol. 7, pp. 367–373, 2001.

[9] L. Birdsey, C. Szabo, and K. Falkner, "Identifying self-organization and adaptability in complex adaptive systems," in *11th IEEE International Conference on Self-Adaptive and Self-Organizing Systems*, 2017.

[10] M. E. A. Whitehouse and K. Jaffe, "Ant wars: combat strategies, territory and nest defence in the leaf-cutting ant *Atta laevigata*," *Animal Behavior*, vol. 51, no. 6, pp. 1207–1217, 1996.

[11] G. M. Leighton, D. Charbonneau, and A. Dornhaus, "Task switching is associated with temporal delays in temnothorax rugatulus ants," *Behavioral Ecology*, vol. 28, no. 1, pp. 319–327, 2017.

[12] C. G. Langton, "Studying artificial life with cellular automata," *Physica D: Nonlinear Phenomena*, vol. 22, no. 1-3, pp. 120–149, 1986.

[13] L. S. Schulman and P. E. Seiden, "Statistical mechanics of a dynamical system based on conway's game of life," *Journal of Statistical Physics*, vol. 19, no. 3, 1978.

[14] H. V. D. Parunak and S. Brueckner, "Entropy and self-organization in multi-agent systems," in *Proceedings of the 2001 International Conference on Autonomous Agents*, August 2001, pp. 124–130.

[15] A. Wissner-Gross and C. Freer, "Causal entropic forces," *Physical Review Letters*, vol. 110, no. 168702, April 2013.

[16] R. P. Mann and R. Garnett, “The entropic basis of collective behavior,” *Journal of the Royal Society Interface*, vol. 12, 2017.

[17] C. Shannon, “A mathematical theory of communication.” *The Bell Systems Technical Journal*, vol. 27, pp. 379–423, 623–656, July, October 1948.

[18] A. Greven, G. Keller, and G. Warnecke, *Entropy*. Princeton University Press, 2003.

[19] M. Mnif and C. Müller-Schloer, “Quantitative emergence,” in *2006 IEEE Mountain Workshop on Adaptive and Learning Systems*, 2006, pp. 78–84.

[20] E. M. Ronald, M. Sipper, and M. S. Capcarrére, “Design, observation, surprise! a test of emergence,” *Artificial Life*, vol. 5, no. 3, pp. 225–239, 1999.

[21] E. Bonabeau, M. Dorigo, and G. Theraulaz, *Swarm Intelligence: From Natural to Artificial Systems*. Oxford University Press, 1999.

[22] A. Rajasekhar, N. Lynn, S. Das, and P. Suganthan, “Computing with the collective intelligence of honey bees – a survey,” *Swarm and Evolutionary Computation*, vol. 32, pp. 25 – 48, 2017.

[23] N. Razin, J.-P. Eckmann, and O. Feinerman, “Desert ants achieve reliable recruitment across noisy interactions,” *Journal of the Royal Society Interface*, vol. 10, no. 82, 2013.

[24] M. Haghnevis, “An agent-based optimization framework for engineered complex adaptive systems with application to demand response in electricity markets,” Ph.D. dissertation, Arizona State University, 2013.